

Интерактивная сегментация изображений

Лекция 7

Два типа сегментации

- Автоматическая
 - Сегментация производимая без взаимодействия с пользователем
 - Картинка на входе, регионы на выходе
- Интерактивная
 - Сегментация, управляемая пользователем, допускающая и/или требующая ввода дополнительной информации
 - Пример – «волшебная палочка» в Photoshop

Интерактивная сегментация – постановка задачи

- Рассматривается задача разбиения на 2 области - объект и фон
- На вход алгоритм получает:
 - исходное изображение
 - дополнительную информацию от пользователя:
 - ограничение на то, что некоторые конкретные пиксели обязательно должны принадлежать объекту (фону)
 - ограничивающий прямоугольник вокруг объекта
 - примерную границу объекта
- В процессе работы алгоритма пользователь может уточнять или дополнять входные данные.

Пример интерактивной сегментации изображения



Одна из основных проблем - отсутствие единой метрики качества
Субъективное сравнение результатов
Разный пользовательский ввод (объём)

Субъективное сравнение

- Отсегментировать данным алгоритмом все изображения не хуже заданного уровня (сравнение на <не хуже> осуществляется субъективно самими пользователями, для сравнения даётся какой-то образец сегментации). После чего измеряется при каком алгоритме на эту задачу ушло меньше всего времени
- За заданное, ограниченное время, как можно лучше отсегментировать данные изображения. После истечения положенного срока, пользователь субъективно оценивает, какой алгоритм дал лучшую сегментацию.

Тестирование методов на общей базе изображений

Например, Berkeley Segmentation Dataset насчитывает более 1000 изображений, отсегментированных вручную 30 разными людьми.

<https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>

Для чего это используется?

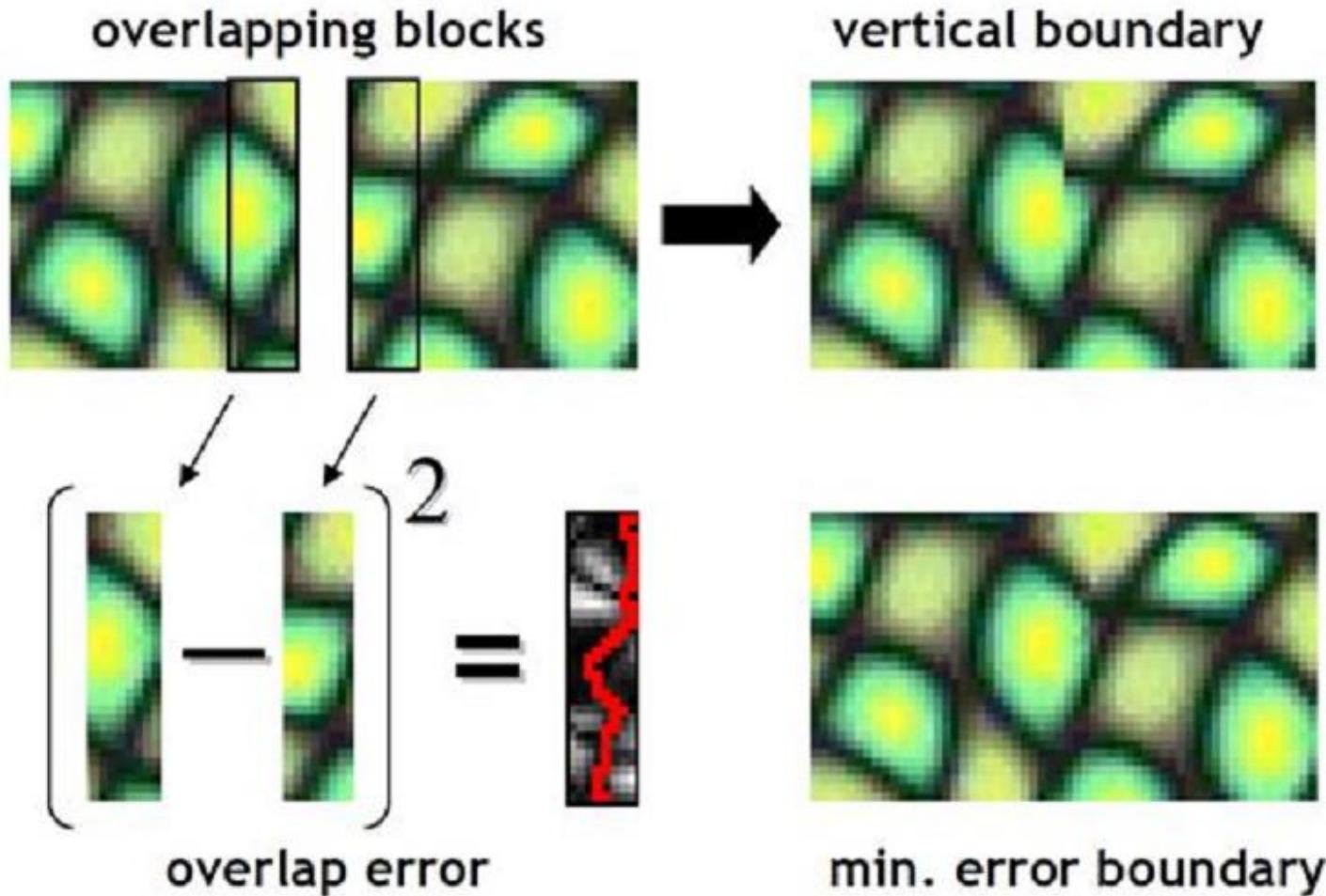
Интерактивная сегментация — это workhorse (рабочая лошадка) для многих задач:

- Фоторедакторы: Быстрое выделение и вырезание объектов, замена фона (например, "Магическая палочка" в Photoshop, но гораздо умнее).
- Разметка данных: Самый главный инструмент для создания датасетов для компьютерного зрения. Без интерактивной сегментации разметка каждого объекта вручную занимала бы дни.
- Медицинская визуализация: Выделение опухолей, органов или других аномалий на КТ, МРТ и УЗИ сценах. Врач может быстро скорректировать результат.
- Робототехника и автономные автомобили: Уточнение сегментации объектов в сложных сценах.
- Дополненная реальность (AR): Для точного наложения виртуальных объектов на реальные.

Применение

- Композиция изображений
- Сшивка
- Создание текстур

GraphCut для сшивки



KWATRA, V., SCHODL, A., ESSA, I., TURK, G., AND BOBICK, A. Graphcut textures: image and video synthesis using graph cuts. SIGGRAPH, 2003

GraphCut для сшивки



Создание текстур



- «Накидываем» исходные экземпляры на изображение и сшиваем их с помощью разрезов графов.
- Перспективное изменение (масштабирование) по необходимости

Синтез текстур в жизни

Bush campaign digitally altered TV ad

President Bush's campaign acknowledged Thursday that it had digitally altered a photo that appeared in a national cable television commercial. In the photo, a handful of soldiers were multiplied many times.

This section shows a sampling of the duplication of soldiers.



Original photograph

Плюсы

- Высокая точность: Позволяет добиться точности, недостижимой для полностью автоматических методов.
- Контроль: Пользователь остается «в контуре управления».
- Эффективность: Значительно ускоряет процесс по сравнению с полностью ручной сегментацией.

Минусы

- Зависит от качества подсказок: "Мусор на входе — мусор на выходе". Неудачные клики могут ухудшить результат.
- Вычислительная стоимость: Современные модели могут требовать значительных ресурсов для инференса в реальном времени.
- Сложные случаи: Может плохо работать с полупрозрачными объектами, сложными текстурами или когда объект и фон очень похожи.

Алгоритмы интерактивной сегментации

Время до 2001

- **Magic wand**
- Intelligent scissors
- Intelligent Paint

После 2001

- Interactive graph cuts
- Lazy Snapping
- GrabCut
- Progressive Cut
- GrowCut

Нейронки

- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Magic Wand (<волшебная палочка>)

- ◆ Идея:
 - Пользователь задает пиксель, запоминается его цвет C_1 ,
 - Все пиксели, цвет которых $\|C - C_1\| < \Theta$ - относятся к выделяемому региону



Эффективность Magic Wand



8 кликов мыши,



более 20 кликов,
причем некоторые части
оленя остались
неотсегментированными

Алгоритмы интерактивной сегментации

Время до 2001

- Magic wand
- **Intelligent scissors**
- Intelligent Paint

После 2001

- Interactive graph cuts
- Lazy Snapping
- GrabCut
- Progressive Cut
- GrowCut

Нейронки

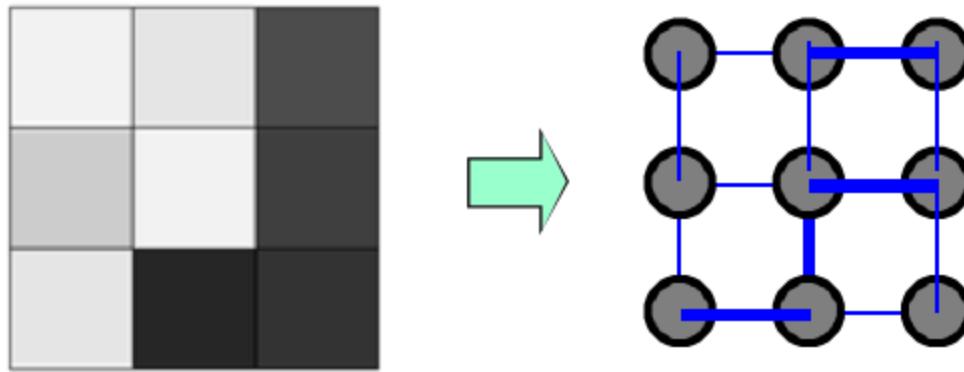
- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Intelligent Scissors (Умные ножницы) [3] 1996



Eric N. Mortensen, William A. Barrett, Intelligent scissors for image composition,
Siggraph 1995

Принцип действия Intelligent Scissors



1. всё изображение трактуется как взвешенный неориентированный граф, каждая вершина которого соответствует пикселию изображения
2. вершины, соответствующие соседним пикселям связываются ребрами
3. на ребрах данного графа определяется весовая функция
4. значение этой функции мало на ребрах, соответствующих потенциальной границе на изображении.

Что учитывает Intelligent Scissors

$$l(p, q) = w_z \cdot f_z(q) + w_G \cdot f_G(q) + w_D \cdot f_D(p, q)$$

$f_z(q)$ учитывает локальные максимумы градиента

$f_G(q)$ учитывает силу градиента (чем он больше, тем f меньше),

$f_D(p, q)$ стимулирует более гладкие границы

Действия пользователя для Intelligent Scissors



- Пользователь указывает 2 точки на границе объекта.
- Граница объекта находится с помощью алгоритма нахождения пути минимальной стоимости в графе
- Алгоритм Дейкстры на основе динамического программирования

Плюсы и минусы Intelligent Scissors

- (+) Существуют очень быстрые реализации алгоритма
- (--) Плохо работает в сильно текстурированных областях
(существует множество альтернативных минимальных путей) .

Модификация Intelligent Scissors

- Использование предварительной пересегментации [4]
- Или использование суперпикселей

Алгоритмы интерактивной сегментации

Время до 2001

- Magic wand
- Intelligent scissors
- **Intelligent Paint**

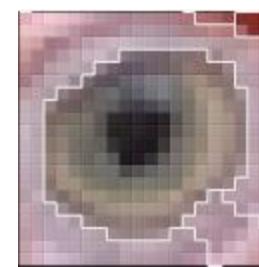
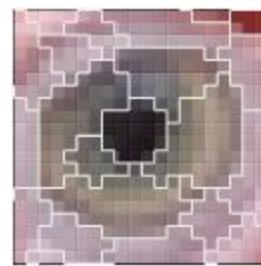
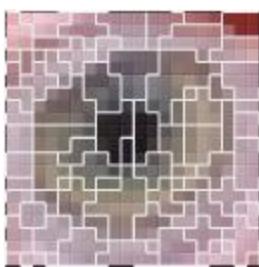
После 2001

- Interactive graph cuts
- Lazy Snapping
- GrabCut
- Progressive Cut
- GrowCut

Нейронки

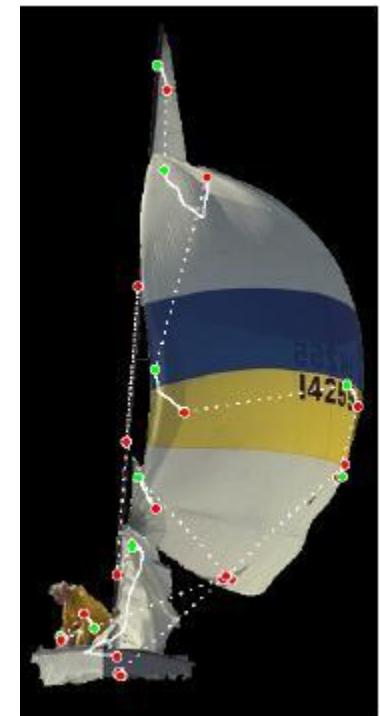
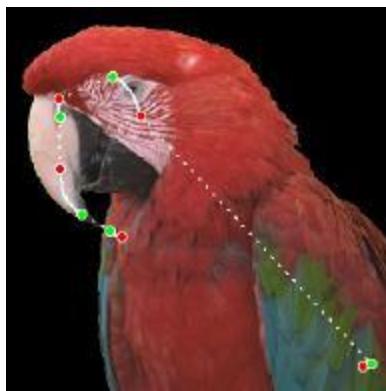
- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Intelligent Paint [5]



- ◆ Идея:
 - Предварительная иерархическая сегментация изображения
 - Анализ «мазков» пользователя для объединения регионов предварительной сегментации

Результаты работы Intelligent Paint



Коммерческие решения (не рассматриваем медицину)

- Magic Wand
(цветовая статистика)



- Magnetic Lasso
(Live Wire, Intelligent scissors)



Алгоритмы интерактивной сегментации

Время до 2001

- Magic wand
- Intelligent scissors
- Intelligent Paint

После 2001

- **Interactive graph cuts**
- Lazy Snapping
- GrabCut
- Progressive Cut
- GrowCut

Нейронки

- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Прорыв в данной области

- произошёл в 2000 г.
- с изобретением Юрием Бойковым и Мари-Пьер Джолли алгоритма GraphCut
- GraphCut стал де-факто эталонным алгоритмом интерактивной сегментации

Действия пользователя GraphCut

- Пользователь указывает жесткие ограничения на сегментацию – пиксели, принадлежащие объекту (O) и пиксели, принадлежащие фону (B)
 - семена объекта и фона



Yuri Boykov and Marie-Pierre Jolly. Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D images. In International Conference on Computer Vision, (ICCV), vol. I, pp. 105-112, 2001.

GraphCut [6]

- всё изображение – граф
- вершины-пиксели + 2 терминальные вершины (исток и сток)
- вес ребра

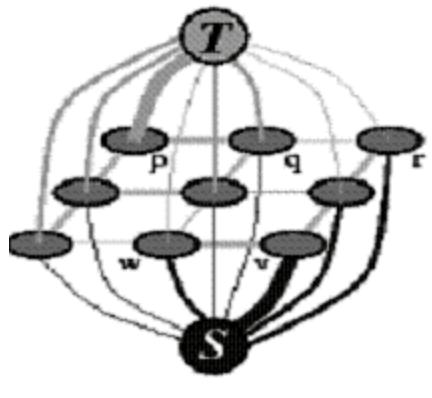
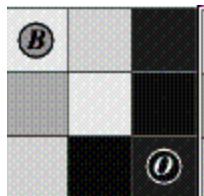
$$B_{p,q} = \exp\left(-\frac{\|C_p - C_q\|^2}{2\sigma^2}\right) * \frac{1}{dist(p, q)}$$

- Формулировка сегментации как минимизации энергии через разрез графа

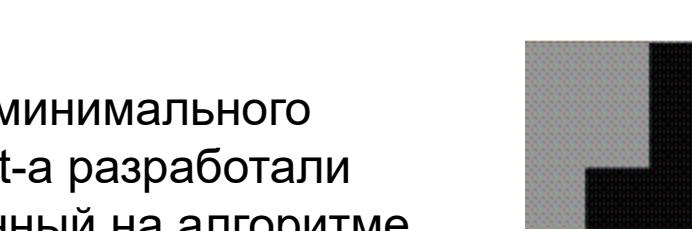
Иллюстрация алгоритма GraphCut

исходное изображение,

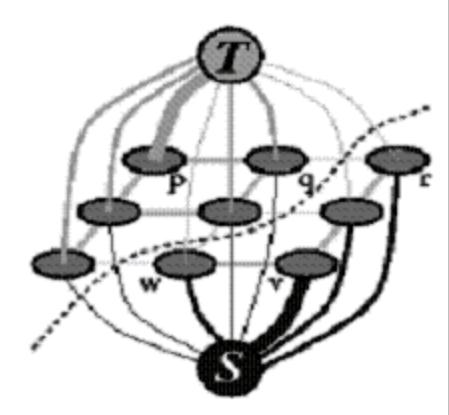
О,В - семена объекта и фона



финальная сегментация



Для нахождения самого минимального разреза авторы GraphCut-а разработали новый алгоритм, основанный на алгоритме Форда-Фалкерсона [7].

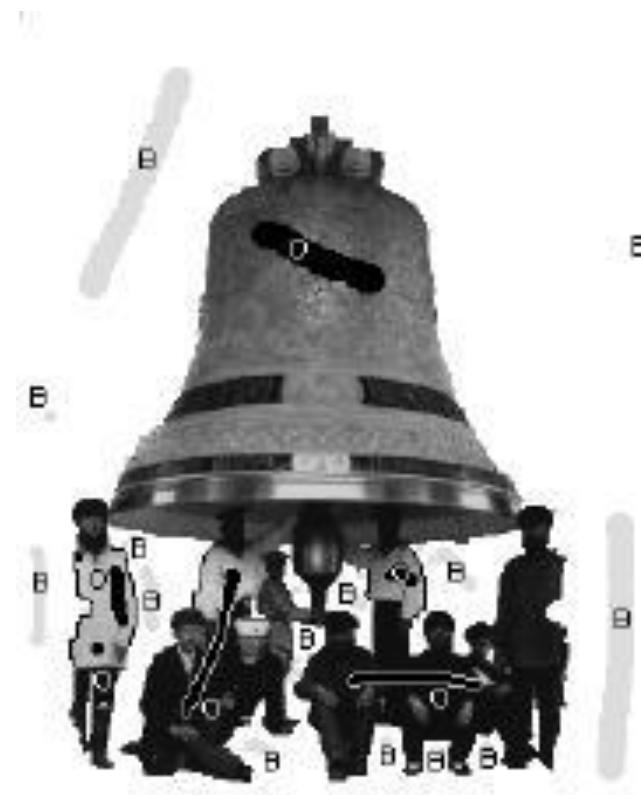


Улучшение алгоритма

ребро	Вес	Для
(p, q)	$B_{(p,q)}$	$(p, q) \in N$
(p, S)	$\lambda * R_p("bkg")$	$p \in P, p \notin O \cup B$
	∞	$p \in O$
	0	$p \in B$
(p, T)	$\lambda * R_p("obj")$	$p \in P, p \notin O \cup B$
	0	$p \in O$
	∞	$p \in B$

цветовая
статистика
объекта и фона,
собранная на
основе
отмеченных
пользователем
семян объекта и
фона

Пример работы алгоритма GraphCut



Алгоритм GraphCut. Примеры



Алгоритмы интерактивной сегментации

Время до 2001

- Magic wand
- Intelligent scissors
- Intelligent Paint

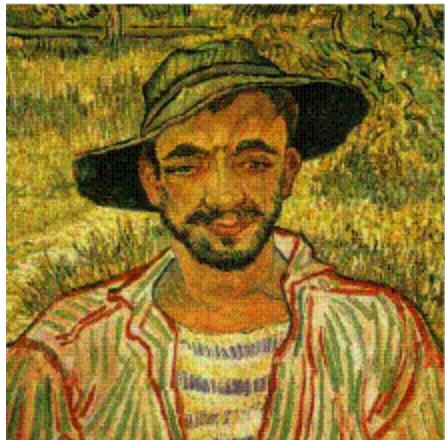
После 2001

- Interactive graph cuts
- **Lazy Snapping**
- GrabCut
- Progressive Cut
- GrowCut

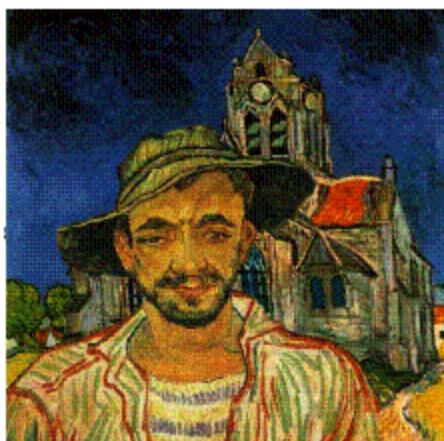
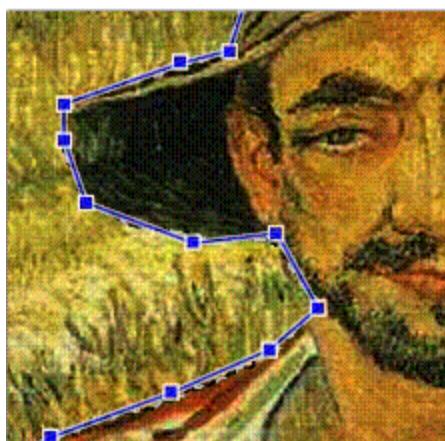
Нейронки

- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Lazy Snapping [11]



(а) - исходное изображение;
(б) - пользователь указывает всего
3 линии (2 внутри объекта, 1 снаружи),
все линии рисуются далеко от границы

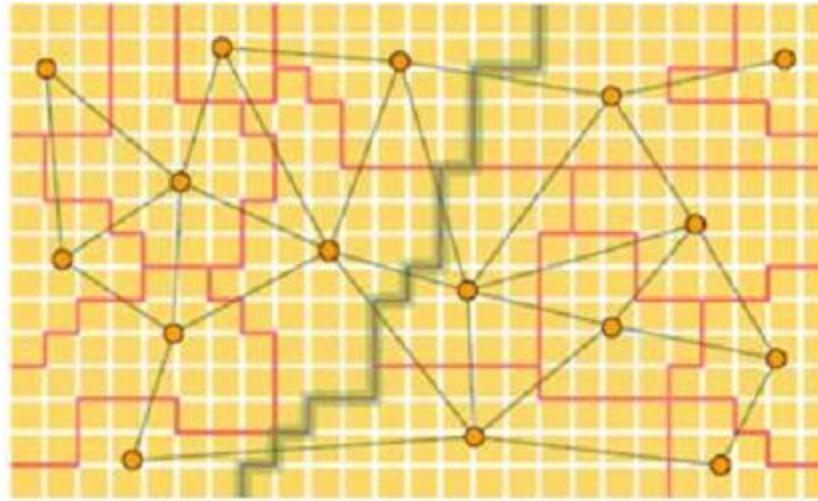


(в) - полигональное редактирование
границы объекта;
(г) - композиция вырезанного
изображения и ещё одной картины
Ван Гога.

То же, что и Interactive GraphCuts, но на карте суперпикселей

Lazy Snapping – особенности

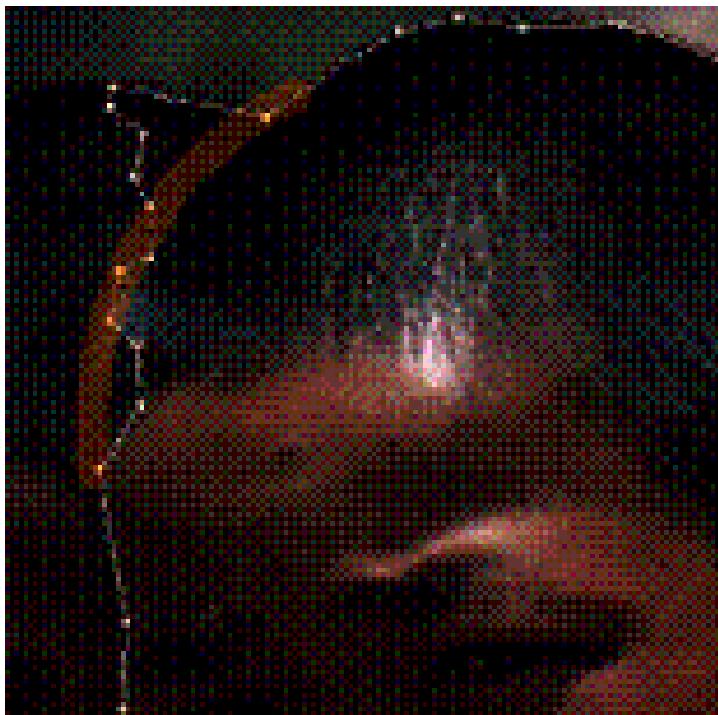
- Карта суперпикселей строится алгоритмом watershed



- В 10 раз быстрее чем на пикселях при сравнимом визуальном качестве
- Учитывается цвет фона и объектов.
- Выборка из размеченных пользователем областей кластеризуется k-средними на 64 кластера, вероятности считаются как отношение близости к объекту/фону

Lazy Snapping - редактирование границы

- получившаяся граница объекта преобразуется в полигон
- для редактирования границы объекта можно:
 - переместить, удалить 1 или несколько вершин полигона, добавить новую вершину;
 - использовать Overriding Brush в неправильно отсегментированной области (начало и конец мазка должны лежать на полигоне)



Результат Lazy Snapping



- Уточнение границ считается как задача поиска оптимального разреза в узкой полосе, отмеченной пользователем

Примеры сегментации с помощью Lazy Snapping



Алгоритмы интерактивной сегментации

Время до 2001

- **Magic wand**
- Intelligent scissors
- Intelligent Paint

После 2001

- Interactive graph cuts
- Lazy Snapping
- **GrabCut**
- Progressive Cut
- GrowCut

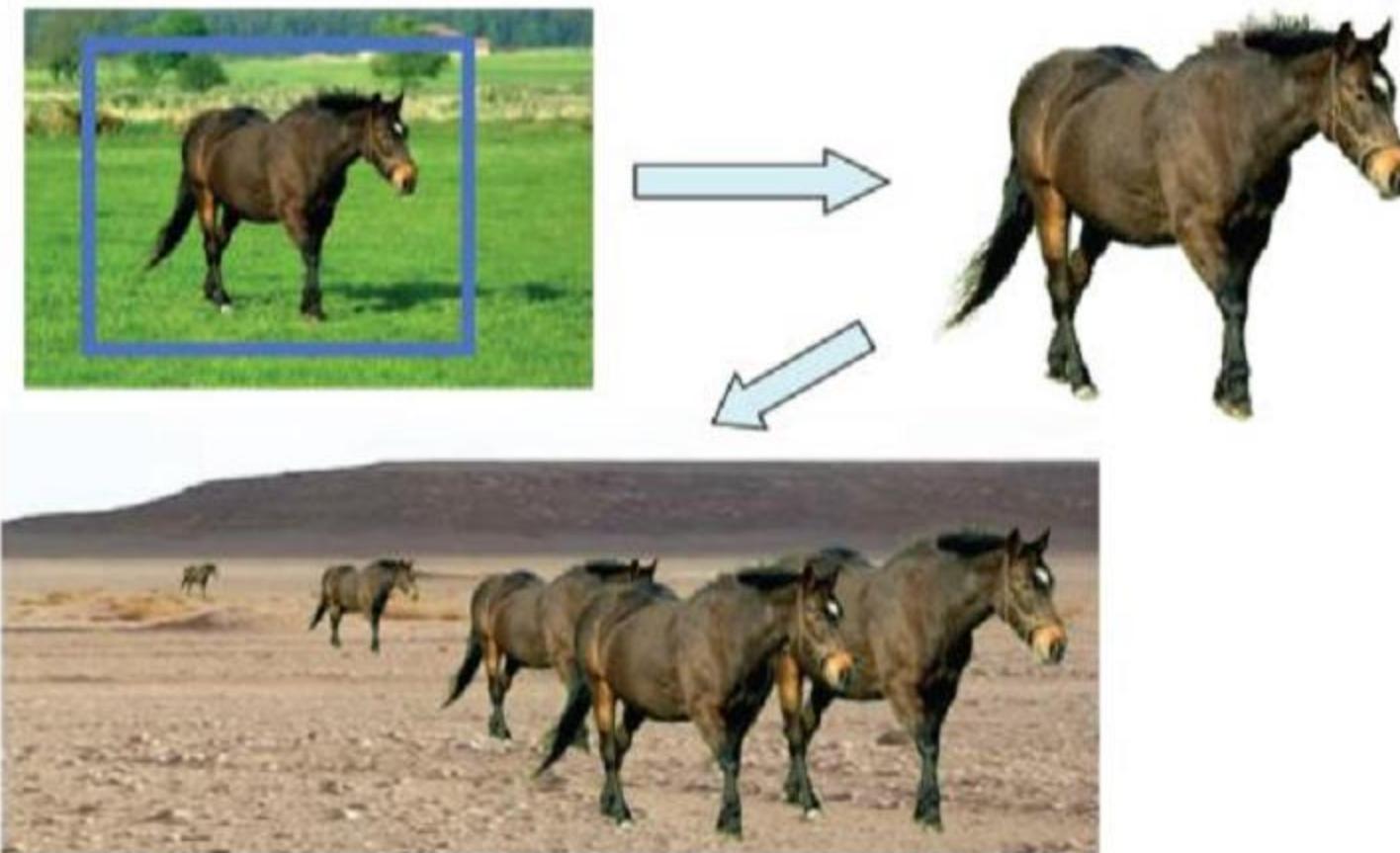
Нейронки

- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Демяненко Я.М. ЮФУ

43

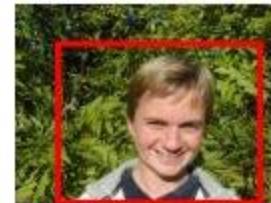
GrabCut [9]



C. Rother, V. Kolmogorov, and A. Blake. Grabcut – interactive foreground extraction using iterated graph cuts. Proc. ACM Siggraph, 2004.

Итеративная схема сегментации GrabCut :

- исходя из цветового распределения внутри и снаружи ограничивающего прямоугольника, строится первая цветовая статистика объекта и фона
- сегментация GraphCut-ом, использующим цветовую статистику
- уточнение цветовой статистики. Граф, в котором ищется минимальный разрез, перевзвешивается





Инициализация пользователя



К-средних для
обучения статистик
цветов



Graph cuts для
сегментации

- Цвет фона и объекта моделируется смесью гауссиан (обычно 5-8 компонент)
- Каждый пиксель объекта/фона сопоставляется компоненте смеси
- Затем уточняются параметры смеси

Slide by C. Rother

Примеры



Slide by C. Rother

Демяненко Я.М. ЮФУ

47

Сложные примеры

Camouflage &
Low Contrast

Initial
Rectangle



Initial
Result



Fine structure



No telepathy



Slide by C. Rother

Сравнение

User
Input

Magic Wand
(198?)



Regions

Intelligent Scissors
Mortensen and Barrett (1995)



Boundary

GrabCut



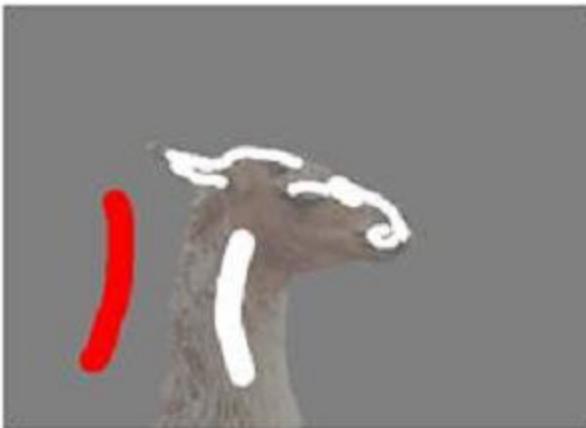
Regions & Boundary

Slide by C. Rother

Сравнение более поздних

Boykov and Jolly (2001)

User Input

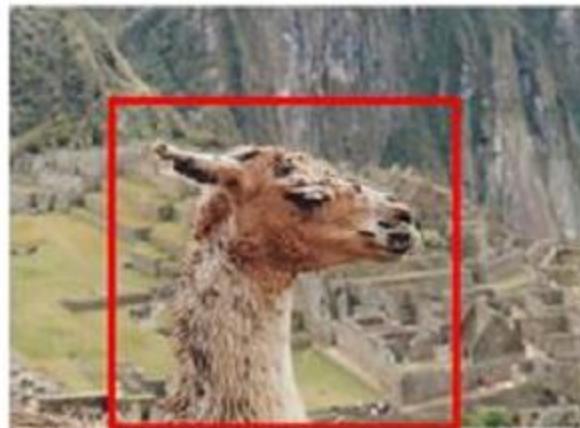


Result

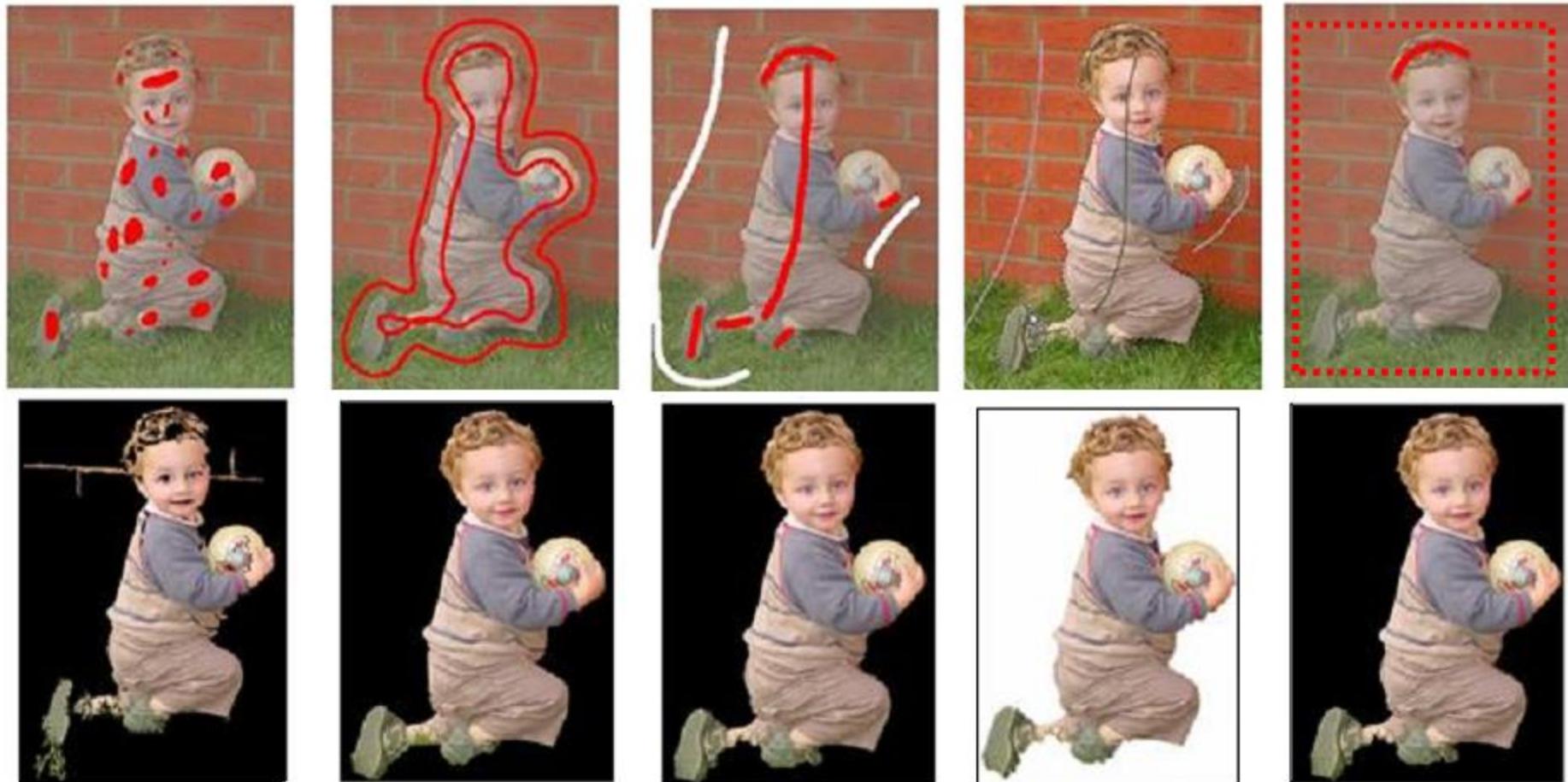


Error Rate: 0.72%

GrabCut



Error Rate: 0.72%



Magic Wand
(198?)

Intelligent Scissors
Mortensen and
Barrett (1995)

Graph Cuts
Boykov and
Jolly (2001)

LazySnapping
Li et al. (2004)

GrabCut
Rother et al.
(2004)

Slide by C. Rother

GrabCut [9]

- GrabCut запатентован компанией Microsoft
- реализован в одной из рабочих версий графического редактора Microsoft Expression [10] (версия V4.0d)

Алгоритмы интерактивной сегментации

Время до 2001

- Magic wand
- Intelligent scissors
- Intelligent Paint

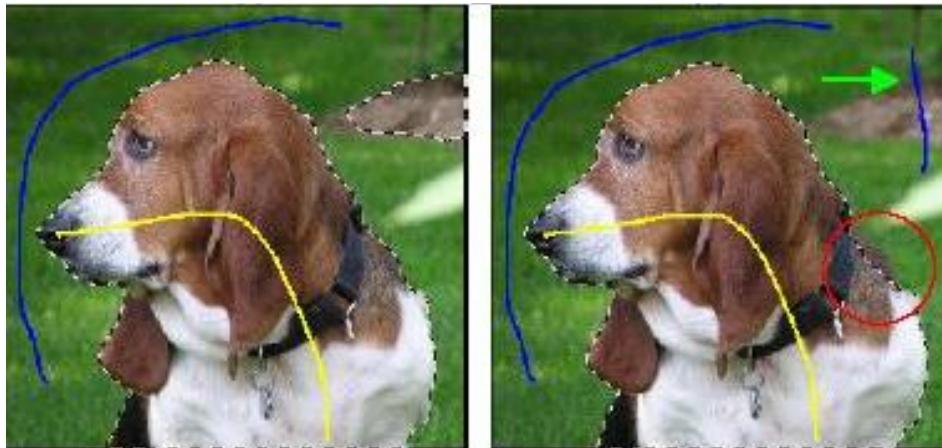
После 2001

- Interactive graph cuts
- Lazy Snapping
- GrabCut
- **Progressive Cut**
- GrowCut

Нейронки

- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Progressive Cut [13]

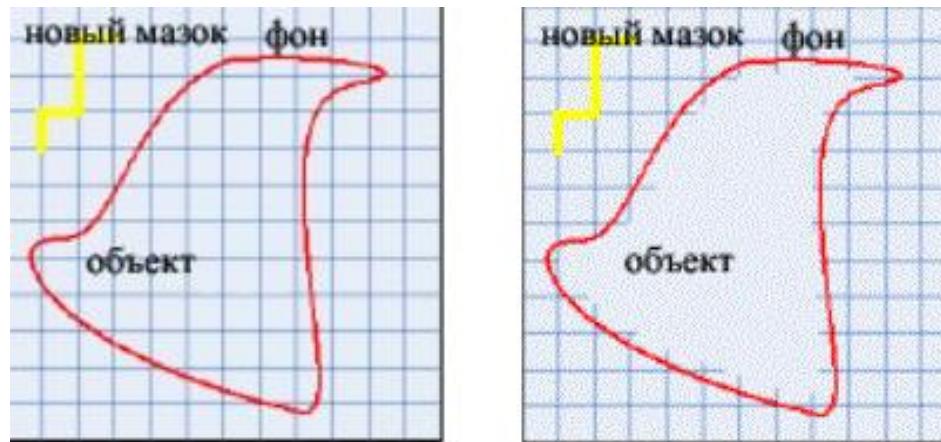


Типичное поведение GraphCut-а

Progressive Cut – намерение пользователя

Progressive Cut пытается понять намерение пользователя, лежащее под тем или иным дополнительным вводом.

1) анализируется, какой тип изменения ожидает пользователь

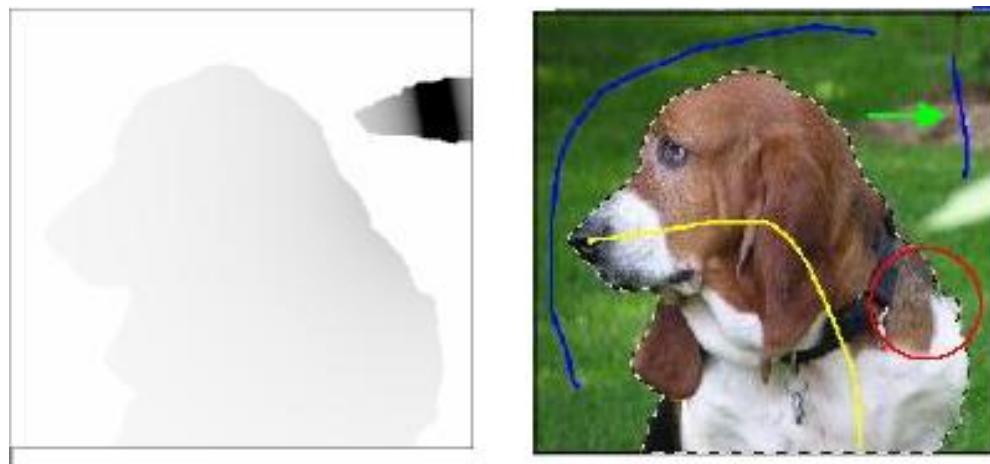


Построение уменьшенного графа на стадии
редактирования изначальной сегментации

Progressive Cut - энергия намерения (intention energy)

2) пользователь обычно ожидает изменения в относительно небольшой области.

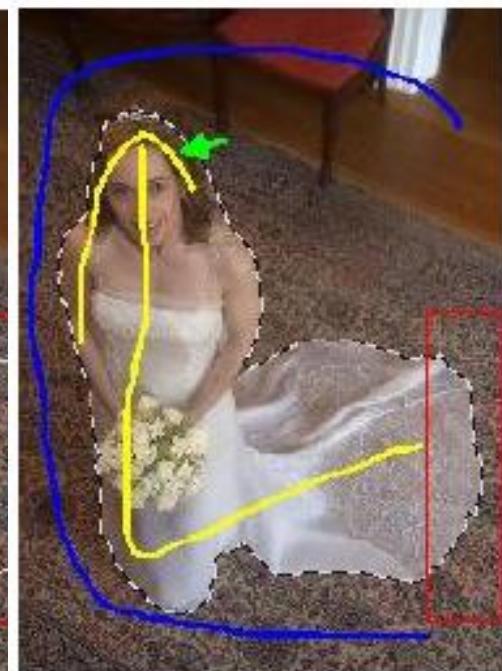
Это записывается в виде дополнительной энергии намерения (intention energy), которая обратно пропорциональна расстоянию от данного пикселя до новых семян



Сравнение редактирования сегментации с помощью GraphCut и Progressive Cut



Сравнение редактирования сегментации с помощью GraphCut и Progressive Cut



Алгоритмы интерактивной сегментации

Время до 2001

- Magic wand
- Intelligent scissors
- Intelligent Paint

После 2001

- Interactive graph cuts
- Lazy Snapping
- GrabCut
- Progressive Cut
- **GrowCut**

Нейронки

- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Параметры GrowCut [17]

- основывается на клеточных автоматах
- состояние каждой клетки - (L, Θ, C) ,
где L - метка ('объект', 'фон', 'неизвестно'),
 Θ $[0,1]$ - сила клетки (помечено – 1, не помечено – 0),
 C - вектор признаков клетки (RGB цвет)

Алгоритм GrowCut

- $g(x)$ – монотонно убывающая функция , изменяющаяся в диапазоне $[0,1]$, которая описывает близость цветов двух пикселей (x обычно является нормой разницы цветов пикселей)
- на каждой итерации каждую клетку P 'атакуют' все её соседи Q
- если $\theta_q * g(\|C_p - C_q\|) > \theta_p$, клетка Q захватывает P

GrowCut. Процесс эволюции клеточного автомата



GrowCut

- Данный алгоритм реализован в виде одноименного плагина к Adobe Photoshop [19]
- легко реализуется на GPU [18]
- Официальный сайт алгоритма
www.growcut.com

Результат GrowCut



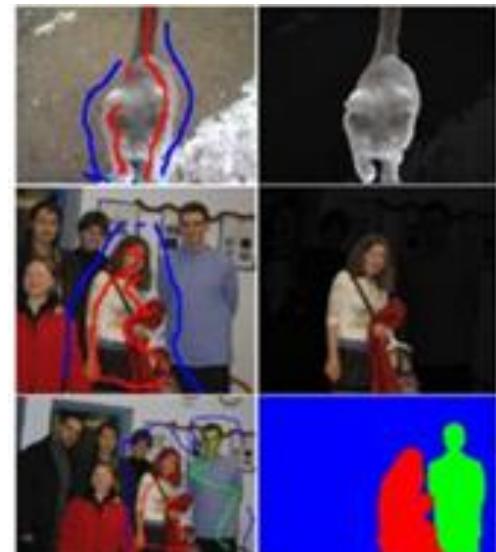
Достоинства и недостатки GrowCut

Плюсы

- прост
- данный алгоритм можно обобщить на случай более 2 сегментов
- время реакции алгоритма близко к нулю
- коррекцию можно выполнить по ходу алгоритма

Минусы

- недостаточно быстро для больших фото
- более рваная граница, чем у Graph Cuts



Развитие GrowCut

В качестве развития данного алгоритма были предложены:

- иерархическая версия, значительно ускоряющая базовый алгоритм,
- а также версия, налагающая на границы итоговых сегментов дополнительные ограничения на гладкость.

Алгоритмы интерактивной сегментации

Время до 2001

- Magic wand
- Intelligent scissors
- Intelligent Paint

После 2001

- Interactive graph cuts
- Lazy Snapping
- GrabCut
- Progressive Cut
- GrowCut

Нейронки

- DeepLab
- U-Net
- HRNet (High-Resolution Net)
- Segment Anything Model (SAM)

Ключевые архитектуры и модели (Исторически и концептуально важные)

U-Net:

Классическая архитектура "encoder-decoder" с skip-connections, которая идеально подходит для задач сегментации, так как хорошо сохраняет мелкие детали.

Классическая U-Net: Основная идея

U-Net была представлена в 2015 году для биомедицинской сегментации. Её название происходит от U-образной формы архитектуры.

Основная философия заключается в сочетании двух процессов:

- Сжатие информации (Encoder): Сеть учится понимать, что изображено на картинке.
- Восстановление пространства (Decoder): Сеть учится точно определять, где находится каждый объект.
- Skip-Connections (Пропускные соединения – «мостики»). Это самый важный элемент U-Net для сегментации

Skip-Connections (Пропускные соединения – «мостики»)

Проблема: При прохождении через энкодер и пулинги теряется информация о мелких деталях и точных границах объектов. Декодер, пытаясь восстановить картинку из сильно сжатого представления, может придумать размытые или неточные границы.

Решение: Skip-connections напрямую соединяют карты признаков из блока энкодера с картами признаками соответствующего блока декодера (который работает с такими же пространственными размерами).

Что дает Skip-Connections?

Передача мелких деталей: Декодер получает прямой доступ к высокоуровневым, но еще не слишком сжатым картам признаков, которые содержат информацию о границах и текстурах.

Объединение контекста и локализации: Энкодер предоставляет контекст (что это за объект), а skip-connections предоставляют локализацию (где его точные границы). Декодер объединяет их для получения точной маски.

Применение U-Net в интерактивной сегментации

Многомодальный ввод (Multi-Modal Input)

Входом в сеть является уже не просто изображение, а объединение нескольких карт.

- Канал изображения: Исходное RGB-изображение (3 канала).
- Каналы взаимодействия пользователя: Создаются дополнительные одноканальные изображения того же размера, что и исходное.

Архитектура кодировщика (Encoder)

Энкодер U-Net учится извлекать признаки не только из самого изображения, но и из комбинации «изображение + пользовательские подсказки». Он кодирует и визуальную информацию, и намерение пользователя в единое представление.

Архитектура декодера (Decoder) и Skip-Connections

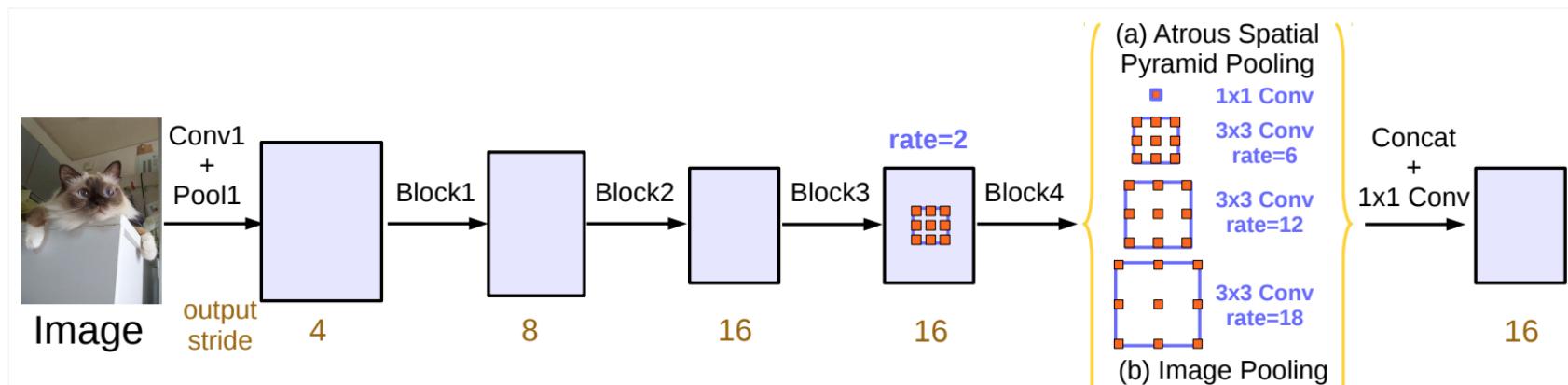
Работает абсолютно аналогично классической U-Net. Именно здесь мелкие детали из энкодера, объединенные с пользовательскими подсказками, используются для «прорисовки» четкой и точной маски объекта, который отметил пользователь.

Ключевые архитектуры и модели (Исторически и концептуально важные)

DeepLab (и его вариации, например, DeepLabv3+):

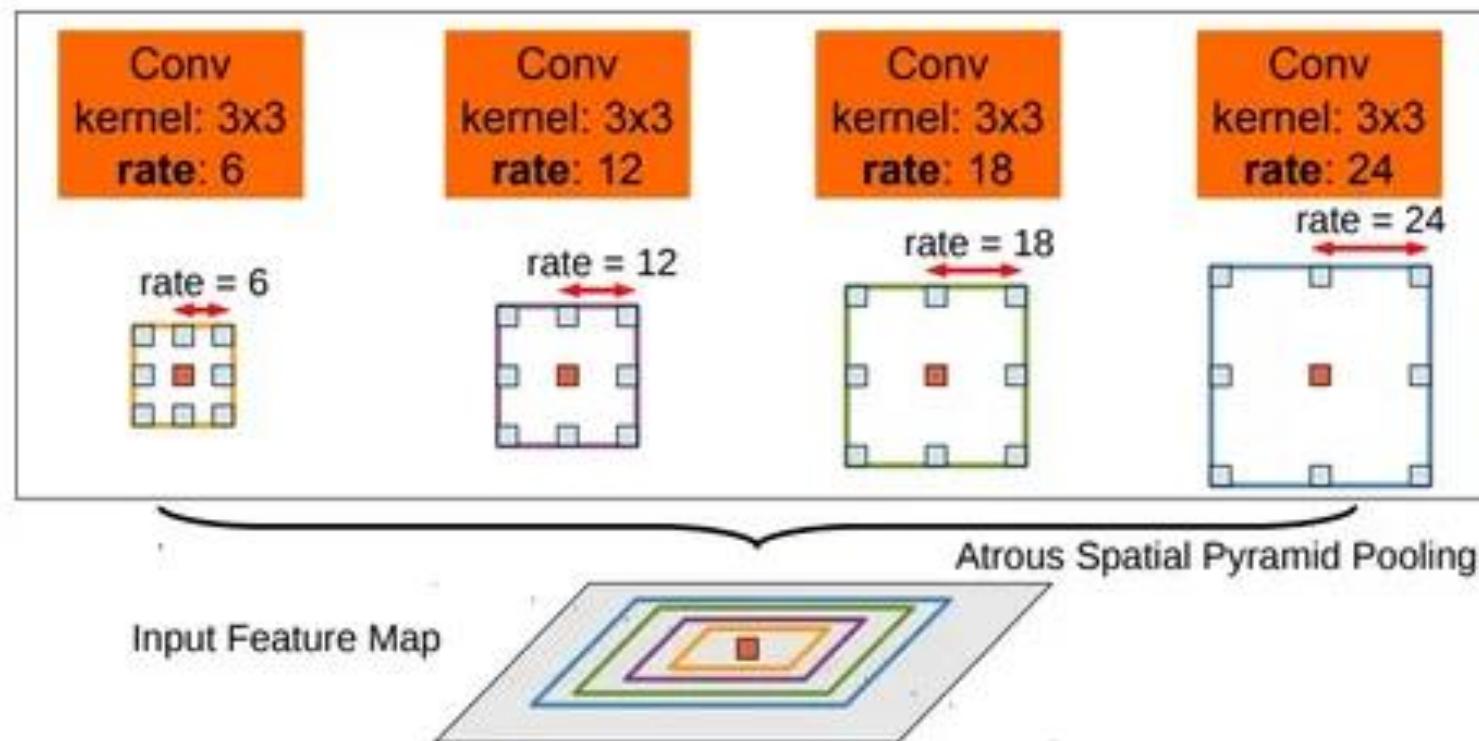
Одна из самых популярных архитектур для семантической сегментации, которая часто используется как backbone (основа) для интерактивных методов.

Использует атрусы-свёртки (ASPP) для захвата контекста на разных масштабах.



Atrous Spatial Pyramid Pooling (ASPP)

это механизм пирамидального свёртывания, который использует несколько параллельных свёрток с разными коэффициентами расширения для захвата объектов разных размеров на изображении.



Типичная структура ASPP (как в модели DeepLabv3+)

- Одна свёртка 1×1 : для захвата информации в исходном масштабе (без контекста).
- Три атрюс-свёртки 3×3 с разными dilation rates (например, 6, 12, 18): для захвата контекста в малом, среднем и большом масштабах.
- Global Average Pooling (Глобальное усреднение): «схлопывает» всё feature map'ы в один вектор, чтобы захватить глобальный контекст всей сцены.

Ключевые архитектуры и модели (Исторически и концептуально важные)

HRNet (High-Resolution Net):

Поддерживает высокое разрешение feature maps на протяжении всей сети, что позволяет очень точно сегментировать границы объектов.

Решение HRNet

Вместо последовательного сжатия и последующего восстановления, HRNet поддерживает высокое разрешение на протяжении всей сети.

Она состоит из нескольких параллельных ветвей:

- Ветвь высокого разрешения: Работает с исходным или близким к исходному разрешением. Отвечает за точное позиционирование и детализацию границ.
- Ветви низкого разрешения: Параллельно создаются ветви, где изображение уменьшено (в 2, 4, 8 раз и т.д.). Эти ветви отвечают за извлечение контекстной, семантической информации.

Эти ветви не изолированы. На каждом этапе сети происходит многократный обмен информацией (fusion) между ветвями разного разрешения. Ветвь высокого разрешения получает семантический контекст от ветвей низкого разрешения, а ветви низкого разрешения «подправляются» за счет точной пространственной информации от ветви высокого разрешения.

Почему HRNet так хорошо подходит?

Сохранение точных границ

Пользователь кликает на границе объекта. Чтобы сегментационная маска точно прошла по этому клику, критически важна информация о пикселях в высоком разрешении. HRNet, не теряя ее, обеспечивает четкие границы.

Учет глобального контекста

Чтобы понять, какой именно объект пользователь хочет выделить (даже если клик стоит на его краю), нужна семантическая информация. Ветви низкого разрешения в HRNet предоставляют эту информацию.

Эффективность и точность

Благодаря параллельной структуре и отсутствию тяжелого декодера, HRNet часто оказывается эффективнее и точнее классических энкодер-декодер сетей для этой задачи.

Ключевые архитектуры и модели (Исторически и концептуально важные)

Segment Anything Model (SAM) от Meta (2023):

SAM — это не просто еще одна модель для семантической сегментации. Это фундаментальная модель (Foundation Model) для компьютерного зрения, обученная на гигантском датасете (SA-1B), которая переосмысливает сам подход к сегментации, делая его интерактивным, гибким и универсальным

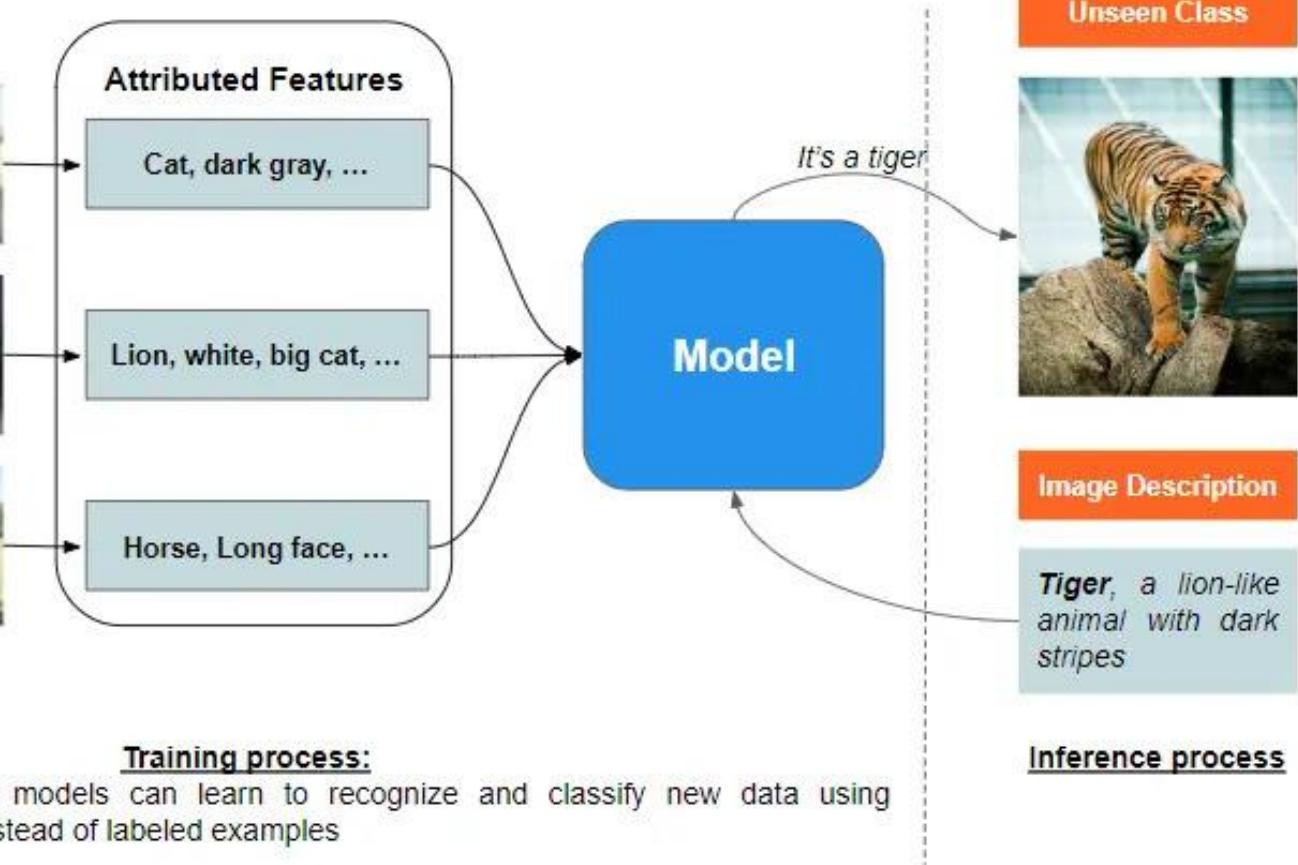
Это настоящий прорыв в этой области.

SAM — это огромная модель, обученная на миллиардах масок. Её ключевая особенность — zero-shot перенос: она может сегментировать объекты, которых никогда не видела во время обучения, на основе одной лишь подсказки (клик, рамка, текст). SAM стал своего рода «фундаментальной моделью» для сегментации.

Seen Classes



Zero-Shot Learning



Ключевые принципы SAM с точки зрения интерактивности

Модель, управляемая промптами (Promptable)

SAM воспринимает любые действия пользователя как «промпты» (подсказки).

«Осознание» объекта (Object-Agnostic)

SAM не знает заранее, что такое «кошка», «машина» или «дерево». Он не обучен на конкретных классах. Вместо этого он научился понимать, что такое «объект» вообще.

Когда вы кликаете на изображение, SAM не ищет «кошку», а смотрит на паттерны пикселей вокруг вашего клика и вычисляет, какие из них, скорее всего, принадлежат одному и тому же целому объекту.

Неоднозначность намерения и множественные валидные маски

Проблема: Один клик на сложном изображении может быть истолкован по-разному.

Решение SAM: Вместо того чтобы угадывать одно-единственное намерение, SAM за один проход генерирует несколько (обычно 3) валидных масок, которые по-разному интерпретируют ваш промпт.

Особенности

Мгновенное предсказание: Модель оптимизирована для работы в реальном времени. Все вычисления происходят за доли секунды, что делает процесс похожим на работу в графическом редакторе.

Мощный кодировщик изображения (Image Encoder): Самая тяжелая часть модели, которая преобразует все изображение в разовое векторное представление. Это позволяет **не пересчитывать** все изображение при каждом новом клике.

Быстрый кодировщик промптов (Prompt Encoder): **Легковесный** модуль, который кодирует клики, рамки и т.д.

Декодер маски (Mask Decoder): **Быстро** «сводит» информацию от кодировщика изображения и кодировщика промптов, чтобы генерировать финальные маски.

Чем SAM революционен для интерактивной сегментации?

Универсальность: Одна модель для всех типов объектов и изображений.

Интуитивность: Интерфейс «кликнул и получил» крайне прост для понимания.

Итеративность: Модель создана для диалога, она позволяет легко и быстро исправлять ошибки.

Скорость: Работа в реальном времени делает процесс комфортным.

Литература 1

2. Martin D., Fowlkes C., Tal D., Malik J. A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. *Proceedings of 8th International Conference on Computer Vision*. 2001. 2. N. 1. P. 416-423.
3. Mortensen, E., and Barrett, W. 1995. Intelligent scissors for image composition. *Proc. ACM Siggraph*, 191-198.
4. Mortensen, E., and Barrett, W. 1999. Tobogan-based intelligent scissors with a four parameter edge model. In *Proc. IEEE Conf. Computer Vision and Pattern Recog.*, vol. 2, 452-458.
5. Reese, L. 1999 Intelligent Paint: Region-Based Interactive Image Segmentation. *Master's thesis*, Department of Computer Science, Brigham Young University, Provo, UT.
6. Y. Boykov and M.-P. Jolly. Interactive organ segmentation using graph cuts. In *Medical Image Computing and Computer-Assisted Intervention*, pages 276-286, 2000.
7. Boykov, Y., and Kolmogorov, V. 2004. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.* 26, 9, 1124-1137.
8. Boykov, Y., and Jolly, M.-P. 2001. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. In *Proc. Of the International Conference on Computer Vision*, vol. 1, 105-112.
9. Rother, C., Kolmogorov, V., and Blake, A. 2004. Grabcut - interactive foreground extraction using iterated graph cuts. *Proc. ACM Siggraph*.

Литература 2

10. http://www.microsoft.com/products/expression/en/graphic_designer/default.mspx
11. Li, Y., Sun, J., Tang, C.-K., and Shum, H.-Y. 2004. Lazy Snapping, *Proc. ACM Siggraph*
12. Vincent, L., and Soille, P. 1991. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-13*, 6 (June), 583-598.
13. Wang, C., Yang, Q., Chen, M., Tang, X., Ye, Z.F. 2006. Progressive Cut. *Proc ACM SIG Multimedia*
14. Grady, L., and Funka-Lea, G. 2004. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. In *ECCV Workshops CVAMIA and MMBIA*, 230-245.
15. Grady, L., Schiwietz, T., Aharon, S., and Westermann, R. 2005. Random walks for interactive organ segmentation in two and three dimensions: Implementation and validation. In *Proceedings of MICCAI 2005*, Springer, Palm Springs, CA, J. Duncan and G. Gerig, Eds., no. 2 in LNCS 3750, MICCAI Society, 773-780.
16. Grady, L. 2005. Multilabel random walker image segmentation using prior models. In *CVPR* (1), 763-770.
17. Vezhnevets, V., and Konouchine, V., 2005. "GrowCut"- interactive multi-label N-D image segmentation by cellular automata, *Proc Graphicon*.
18. <http://www.sonycsl.co.jp/person/nielsen/PT/SHADERX5/shaderx5.html>
19. <http://www.growcut.com>
20. Talbot, J.F., and Xu, X., Implementing Grabcut,
<http://students.cs.byu.edu/~jtalbot/research/Grabcut.pdf>